1. Cloud concepts (6 points)

   1) What is cloud definition made by NIST? (Can you summarize no more than three sentences?) (3 points)

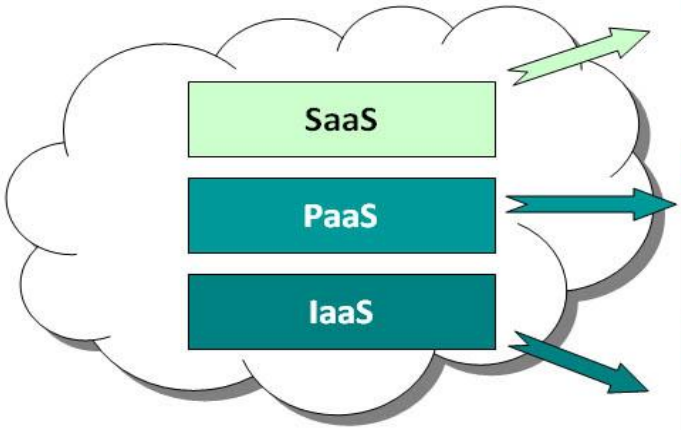
      - Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.

   2) In general, what are three basic cloud service models and what is the relation between them? (3 points)

      a. Software as a Service - SaaS

      b. Platform as a Service – PaaS

      c. Infrastructure as a Service – IaaS



Who Uses It	What Services are available	Why use it?
Business Users	EMail, Office Automation, CRM, Website Testing, Wiki, Blog, Virtual Desktop ...	To complete business tasks
Developers and Deployers	Service and application test, development, integration and deployment	Create or deploy applications and services for users
System Managers	Virtual machines, operating systems, message queues, networks, storage, CPU, memory, backup services	Create platforms for service and application test, development, integration and deployment

2. Virtualization techniques (21 points)

   1) What is *process virtual machine*? What is *system virtual machine*? (5 points)

      - *process virtual machine* --capable of supporting an individual process

      - *system virtual machine* --provide a complete system environment

- can support a "guest OS" with (probably) many user processes

2) What is *full virtualization*? What is *paravirtualization*? (6 points)

- **full virtualization** is a **virtualization** technique used to provide a certain kind of virtual machine environment, namely, one that is a complete simulation of the underlying hardware.

- **Paravirtualization** is an enhancement of virtualization technology in which a guest OS is recompiled prior to installation inside a virtual machine. **Paravirtualization** allows for an interface to the virtual machine that can differ somewhat from that of the underlying hardware.

3) What are type-1 hypervisor and type-2 hypervisor? (4 points)

- type-1 --**Bare-metal**
  - Has complete **control over hardware**
  - Does not have to **"fight" an OS**

- type-2-- **Hosted**
  - Avoid **code duplication**: need not code a **process scheduler**, **memory management** system – the **OS already does** that
  - Can run native **processes alongside** VMs
  - Familiar environment – **how much CPU** and **memory** does a VM take? Use **top**! How big is the **virtual disk**? **ls –l**
  - Easy management – stop a VM? Sure, just kill it!

4) Compare two types of I/O virtualization techniques (Emulation vs. Paravirtualization) and describe their pros/cons. (6 points)

- Emulation --Hypervisor implements **virtual NIC** (by the specification of a real NIC, e.g., Intel, Realtek, Broadcom)
  NIC **registers** (X, Y, Z, T, R, …) are just **variables** in hypervisor (host) **memory**
  If **guest writes** '1' to **register T**, **hypervisor reads** buffer from **memory [X,X+Y)** and passes it to **physical NIC** driver for transmission
  When physical NIC interrupts (**TX complete**), hypervisor **injects** TX complete interrupt into guest

  Pro:
  - **Unmodified guest** (guest already has drivers for Intel NICs…)

Cons:

- **Slow –** every access to every NIC register causes a **VM exit** (trap to hypervisor)
- Hypervisor needs to **emulate complex hardware**

- Paravirtualization
  Add virtual NIC driver into guest (**frontend**)
  Implement the virtual NIC in the hypervisor (**backend**)
  Everything works just like in the emulation case…
  …except **– protocol** between frontend and backend

  Pro:
  - **Fast –** no need to emulate physical device

  Con:
  - Requires **guest driver**

3. Distributed Systems and Distributed Computing (8 points)
   1) What is a distributed system? List at least five characteristics of distributed systems. (5 points)

      - The Collection of individual computers that appears to its users as a single coherent system.

      a. Scalability
      b. Support Heterogeneity
      c. Continuous Availability
      d. Users are unaware about the distribution and heterogeneity of the system
      e. The components of a system are also unaware of the heterogeneity
      f. Users and applications can interact with the system uniformly

   2) Describe one classification in term of hardware for distributed systems. (2 points)

      a. Multi Processor Systems.
      b. Multi Computer Systems.

4. Cluster Computing (9 points)

   1) What is a computer cluster? What is cluster computing? List three types of Clusters. (6 points)

      - A computer cluster is a group of tightly coupled computers that work together closely so that it can be viewed as a single computer.

      - A group of interconnected <u>WHOLE COMPUTERS</u> works together as a unified computing resource that can create the illusion of being one machine having parallel processing.

- High-availability (HA)
  Load-balancing
  High- Performance(HP)

2) Why should we choose Clusters than single 1's for some services? (3 points)

- Price/Performance
  The reason for the growth in use of clusters is that they have significantly reduced the cost of processing power.

- Availability
  Single points of failure can be eliminated, if any one system component goes down, the system as a whole stay highly available.

- Scalability
  HPC clusters can grow in overall capacity because processors and nodes can be added as demand increases.

5. Parallel Computing (6points)

What is parallel computing? Why should we use parallel computing? Describe two types of parallel computing.

- Parallel computing is the simultaneous use of multiple compute resources to solve a computational problem.

- Saves time
  Solve larger problems
  Cost savings
  Provide concurrency

- Data Parallelism
  Task Parallelism

6. Data Center (6 points)

1) What are the three layers of the conventional topology in a data center? (3 points)

a) Access layer with Top of the Rack (ToR) switches
b) Aggregation layer
c) Core layer

2) What are the key elements of a virtual data center? (3 points)

a. Virtual machine
b. Virtual switches

c. Virtual links

7. OpenStack (20 points)

1) What are the seven core components of OpenStack? Please draw the conceptual architecture first and then describe the functionalities of each component? (14 points)

a. Compute-- Compute (codenamed "Nova") provides virtual servers upon demand. Compute stores and retrieves virtual disks ("images") and associated metadata in the Image Store ("Glance").

b. Object Storage-- Object Store (codenamed "Swift") allows you to store or retrieve files (but not mount directories like a fileserver).

c. Identity-- Identity (codenamed "Keystone") provides authentication and authorization for all the OpenStack services.

d. Dashboard-- Dashboard (codenamed "Horizon") provides a modular web-based user interface for all the OpenStack services

e. Block Storage-- Block Storage (codenamed "Cinder") provides persistent block storage to guest VMs.

f. Network-- Network (which used to named "Neutron" but is in the process of being renamed due to a trademark issue) provides "network connectivity as a service" between interface devices managed by other OpenStack services (most likely Nova).

g. Image Service-- Image Store (codenamed "Glance") provides a catalog and repository for virtual disk images. These disk images are mostly commonly used in OpenStack Compute.

2) There are different subcomponents in each main component within OpenStack. Please list at least three subcomponents which are designed mainly for the different scheduling work, and then describe their main functionalities (6 points).

Compute - **nova-scheduler**

- The nova-scheduler process is conceptually the simplest piece of code in OpenStack Nova: take a virtual machine instance request from the queue and determines where it should run (specifically, which compute server host it should run on).
- In practice, it is now one of the most complex.

Block Storage

- Much like nova-scheduler, the **cinder-scheduler** daemon picks the optimal block storage provider node to create the volume on.

Network (quantum-server)

- **quantum-server** accepts API requests and then routes them to the appropriate quantum plugin for action.

8. Software defined network (12 points)

1) What are three tiers in SDN architecture? Please draw the architecture first and then describe the functionalities of each tier. (6 points)

a. Data Plane Tier

Packet forwarding (as per flow table), packet manipulation (as per flow table), statistics collection

b. Control Plane Tier (Network OS)

Data plane resource marshaling, common libraries (e.g., topology, host metadata, state abstractions)

c. Application Tier

Virtual network overlays, network slicing (delegation), tenant-aware broadcast, application-aware path computation, integration with other software packages, policy, security, traffic engineering.

2) Describe how OpenFlow implements the SDN concept? (8 points)

a. OpenFlow is a Layer 2 communications protocol that gives access to the forwarding plane of a network switch or router over the network.

b. OpenFlow enables controllers to determine the path of network packets through the network of switches.

c. The OpenFlow protocol is layered on top of the Transmission Control Protocol (TCP), and prescribes the use of Transport Layer Security (TLS).

d. Controllers should listen on TCP port 6653 for switches that want to set up a connection. Earlier versions of the OpenFlow protocol unofficially used port 6633.

e. OpenFlow allows remote administration of a layer 3 switch's packet forwarding tables, by adding, modifying and removing packet matching rules and actions.

9. Suppose you want to establish a private cloud in CSUF and provide IaaS for students and teachers. (12 points)

You have limited budgets can purchase a few servers and switches (e.g., 3 high performance

servers and each server has 64G memory and 1T storage, and two high-speed switches). Your design goal is to provide remote desktop access for users (support 100 Linux VMs and or 30 Windows VMs). The existing Openstack design is too clumsy in terms of resource consumption and too complicate to set
up your services, and thus you do **NOT** want to use openstack and want to design the system from the scratch.

1) Describe your design requirements to enable the above desired system. (hint: The design requirements will lead you to the system physical and software components design. Thus provide detailed design requirements including networking, storage, computation, access control, monitoring, etc.).

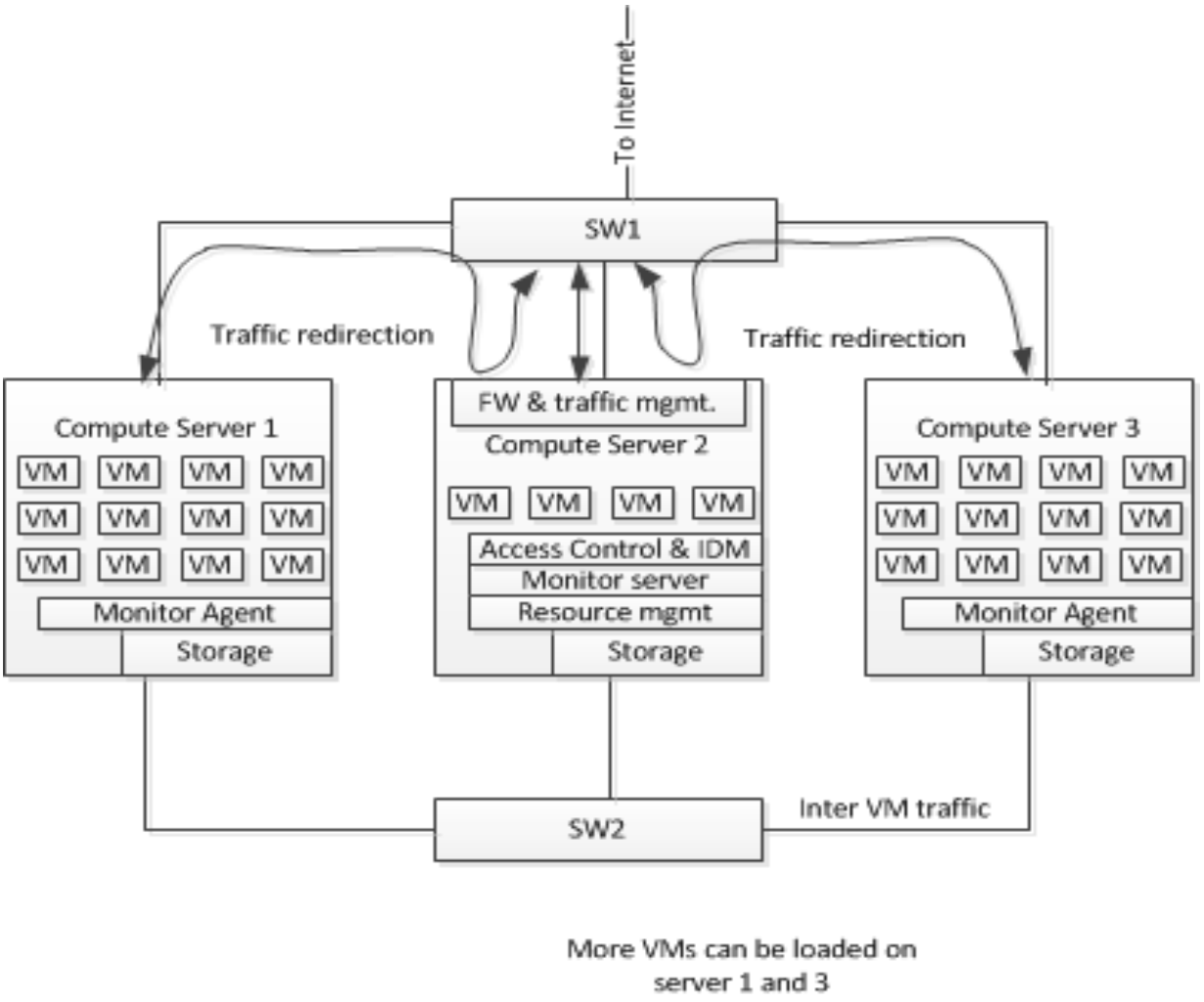
   The answer should be flexible but not restricted to the following answer.

   2.1 The system restrictions are: 3 servers (1TB storage, 64G Ram), 2 SWs.

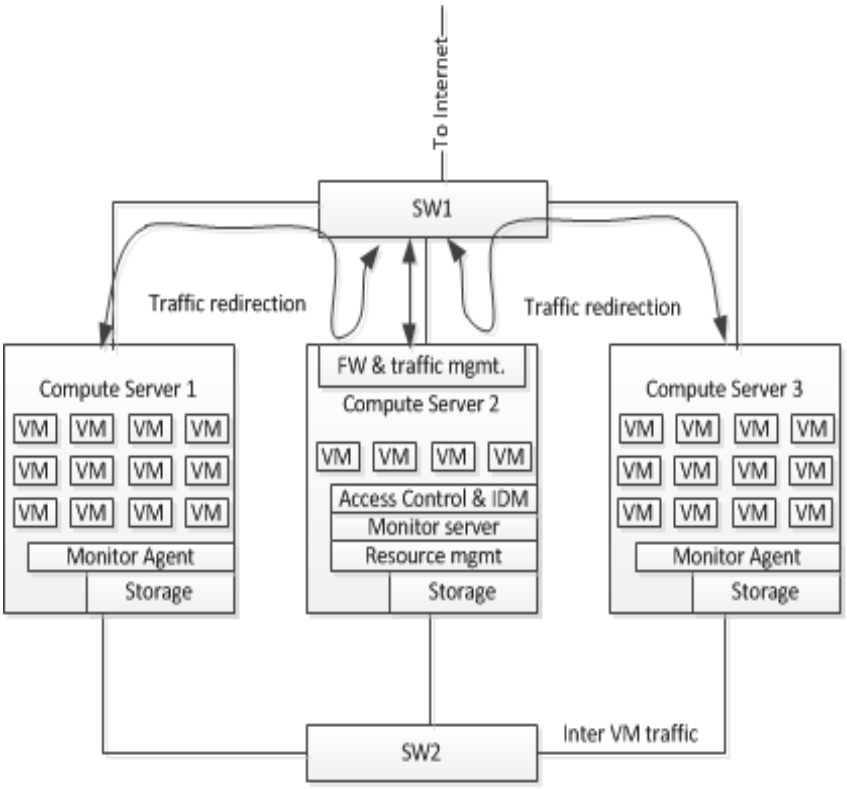
   In order to provider IaaS services for 30 Windows VMs and 100 Linux VMs, assume each Windows requires 2 GB (Thus total 60GB), and each Linux requires 1GB (Thus total 100GB). The total required memory will be 160GB. The total available Ram is 64*3=192GB. Thus, we need to utilize all the servers to provide the VM hosting.

   - Networking: we can use one sw to connect all the servers to the Internet, and one sw is used for inter-vm communications. One of the server need to serve as the GW to route traffic to other servers. A simple approach is to have this server with public IP and redirect the traffic to other servers from/to Internet.
   - Storage: Total available storage is 3T accumulated based on three servers, If each VM require1 100G, thus, we should have sufficient storage to host VM locally on each server.
   - Computation: Each server will serve as a computing node to host VMs. In order to share the workload equally, we should allocate windows and Linux VMs evenly on each compute nodes. Since more control functions will be loaded on one server (e.g., server 2), them more VMs can be loaded on server 1 and server 3.
   - Access control: we need to use one server to serve as the IDM for user management and resource management for computing and networking resource management (e.g., allocate VMs and set up virtual networks, etc).
   - Monitoring: performance monitoring agents should be enabled on each server to monitor the system performance. A monitoring server must be running to demonstrate the system performance.

2) Describe your physical system setup (draw the system topology and illustrate why you want to setup in that way, i.e., pros and cons. Hint: provide your answer according to your provided system design requirements)



3) Describe what software functions that you want to establish on each physical machine, and their interrelations. (Describe carefully about the system components and how they provide you the desired features. Hint: provide your answer according to you provided system design requirements).



Server 1
1. Virtualization software
2. Cloud resource management (controller)
3. storage (objective, volume) software
4. monitoring agent

Server 2
1. Virtualization software
2. IDM
3. Network controller (firewalls, traffic redistributor)
4. storage (objective, volume) software
5. monitoring agent

Server 3
1. Virtualization software
2. Monitoring server
3. storage (objective, volume) software
4. monitoring agent

More VMs can be loaded on server 1 and 3